International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 9, Issue 3, pp 13-27, March-2025

https://doi.org/10.47001/IRJIET/2025.903003

Design and Implementation of an Advanced Web-Based Academic Advising System for Optimizing Student Support and Engagement

[1]Umeh, Mary Rose Ngozi, [2]Anusiuba, Overcomer Ifeanyi Alex, [3]Faseki, Ngozi Francisca, [4]Davies, Naomi Nmesoma

[1,2,4]Department of Computer Science, Faculty of Physical Sciences, Nnamdi Azikiwe University, Awka, Nigeria
[3]Department of Computer Science, Faculty of Computing, Nigeria Army University, Biu Borno State, Nigeria
Authors E-mail: [1]nm.umeh@unizik.edu.ng, [2]oi.anusiuba@unizik.edu.ng, [3]fasekiebenezer@gmail.com, [4]daviesnaomi005@gmail.com

Abstract **- The Web-Based Student Academic Advising System was developed to address the limitations inherent in traditional academic advising methods, which often rely on manual processes that are both time-intensive and difficult to scale to accommodate large student populations. The primary aim of this project is to establish a comprehensive, user-friendly platform that facilitates appointment scheduling between students and advisors, enables the provision of academic guidance, and disseminates pertinent institutional announcements. In addition, the platform empowers academic advisors to monitor and evaluate student performance through data-driven insights, thereby enabling the delivery of personalized academic counsel. The system also incorporates essential functionalities for managing academic records and offering course recommendations, critical components for optimizing student academic pathways. The design and development of the system adhere to the Object-Oriented Analysis and Design Methodology (OOADM), which emphasizes modularity, scalability, and maintainability, ensuring that the architecture of the system is both robust and adaptable to future needs. The key technologies employed in the system's development include React.js and Tailwind CSS for front-end design, coupled with Node.js and Firebase for back-end services, thereby ensuring a responsive, reliable, and efficient user experience. This system's implementation serves to underscore the potential for enhancing student engagement and fostering academic success. By integrating contemporary web technologies, it ensures operational efficiency, scalability, and an enhanced user-centered experience, which collectively contribute to the advancement of academic advising practices within higher educational institutions.**

Keywords: Web-Based Student, Academic Advising System, Student Academic Advising, OOADM.

I. BACKGROUND OF THE STUDY

In an era marked by rapid technological advancements and digital transformation, the landscape of education is experiencing profound shifts. Educational institutions are increasingly adopting technology to enhance various dimensions of the learning experience, including academic advising (Kalsbeek, 2006). Academic advising serves a critical function in supporting students throughout their educational trajectory, providing essential guidance and resources to navigate the complexities of academic requirements and career aspirations (Kuh, 2008).

The role of academic advising is integral to student success and retention in higher education institutions. In light of the rising incidence of poor academic performance in contemporary educational settings, the establishment of a robust academic advising system is indispensable for fostering improved academic outcomes. Such systems support efficient, productive learning environments and contribute significantly to student success. (Anusiuba et al, 2021). Academic advising remains a fundamental activity within educational institutions, offering vital support in career exploration, academic discipline choices, and in addressing challenges such as peer pressure (Tinto, 2012).

Traditionally, academic advising has been conducted through face-to-face interactions between students and advisors. While these interactions have provided substantial value, they are often constrained by scheduling conflicts, geographical limitations, and resource availability. In today's fast-paced, interconnected world, there is an increasing demand for more flexible, accessible, and efficient advising solutions that cater to the diverse needs of both students and advisors (Bates & Sangra, 2011).

An automated and fully functional academic advising system has the potential to be a transformative tool for both students and advisors. The complexity of modern academic curricula, particularly with recent shifts toward general

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 9, Issue 3, pp 13-27, March-2025

https://doi.org/10.47001/IRJIET/2025.903003

education and evolving degree requirements, presents ongoing challenges for faculty advisors to stay current with the latest information. This highlights the pressing need for technology-driven solutions that can streamline and enhance the advising process.

Given the advancements in technology, it is crucial to modernize the educational counseling system, as the current manual procedures are increasingly inadequate to meet the needs of a diverse and expanding student population. The proposed system aims to significantly improve student academic performance by providing a comprehensive, automated platform that bridges the information gap, thereby facilitating better access to academic resources and supporting the development of academic skills. (Anusiuba, 2024) (Anusiuba, Nweke and Egbo, 2019).

This system represents an innovative approach to addressing these challenges and harnessing technology to revolutionize the advising process. By integrating features such as online appointment scheduling, degree planning tools, interactive course catalogs, and real-time communication capabilities with advisors, this platform aspires to create a dynamic, interactive environment. It empowers students to take ownership of their academic journeys while equipping advisors with the necessary tools and resources to offer personalized, data-driven guidance and support.

1.1 Statement of the Problem

Despite the acknowledged benefits of academic advising, many students continue to experience challenges in obtaining timely and personalized guidance (Anusiuba, 2024). Traditional advising methods, which predominantly rely on face-to-face interactions, are often hindered by issues of convenience and time constraints for both students and advisors. Furthermore, the growing complexity of degree requirements and academic pathways necessitates a more dynamic and flexible advising model (Selwyn, 2013).

Consequently, there exists a significant need for a comprehensive, web-based advising system capable of providing academic advice and offering real-time counseling to a large cohort of students concurrently, irrespective of time and location. This study proposes a novel smart advising system designed to address these needs. The system is intended to enhance students' academic performance by delivering individualized advice related to course selection, addressing personal issues that influence academic outcomes, and contributing to overall improvements in academic achievement. By leveraging advanced technological frameworks, the system aims to optimize the advising process, thereby supporting students in navigating their academic trajectories more effectively.

1.2 Aim and Objectives of the Study

The objective of this study is to design and develop a web-based student academic advising system that will provide tailored academic guidance to students. The specific aims of this project are as follows:

1. To conduct a comprehensive review and analysis of existing academic advising systems, identifying key limitations and opportunities for enhancement.
2. To design a user-centered, intuitive web-based platform that effectively addresses the needs of both students and advisors, ensuring secure access to academic information, scheduling functionalities, communication channels, and academic planning tools.
3. To develop the requisite software components and functionalities that support the advising process, including but not limited to appointment scheduling, degree planning, and real-time communication tools.
4. To assess the usability, functionality, and overall effectiveness of the developed system through systematic user testing and the collection of feedback, enabling iterative improvements to enhance system performance and user satisfaction.

II. REVIEW OF RELATED LITERATURE

Academic advising is a comprehensive, structured process aimed at assisting students in achieving their educational and career objectives. It involves a collaborative relationship between advisors and students, where advisors guide students through course selection, understanding academic policies, and accessing campus resources. The foundation of academic advising lies in developmental theory, which focuses on the growth and development of individual students. Key theorists such as Arthur Chickering and Nevitt Sanford have significantly contributed to understanding student development within higher education (Chickering & Reisser, 1993; Sanford, 1966). Many theoretical frameworks underpin effective academic advising practices. Chickering's Seven Vectors of Identity Development identifies seven critical areas of student growth, including developing competence, managing emotions, fostering autonomy, and establishing identity. Advisors support these stages by providing personalized guidance tailored to individual needs (Chickering & Reisser, 1993). Sanford's Theory of Challenge and Support posits that student development is optimized when there is a balance between challenge and support. Advisors must offer challenges that encourage growth while ensuring adequate support to prevent students from becoming overwhelmed (Sanford, 1966). Tinto's Theory of Student Retention highlights the importance of both academic and social integration for student persistence. Effective advising

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 9, Issue 3, pp 13-27, March-2025

https://doi.org/10.47001/IRJIET/2025.903003

fosters a sense of belonging and engagement, key factors in enhancing retention and academic success (Tinto, 1993).

Several models of academic advising are grounded in distinct theoretical perspectives and practical approaches.

1. **Prescriptive Advising** is based on a directive approach, where advisors offer specific recommendations and instructions to students, similar to the traditional medical model. While it is efficient, this model is often criticized for limiting student autonomy and engagement in the advising process (Crookston, 1972).
2. **Developmental Advising**, rooted in developmental theory, emphasizes the advisor-student relationship and focuses on the holistic development of the student. This model promotes collaboration, encouraging students to take an active role in their educational and personal growth (Crookston, 1972).
3. **Intrusive Advising**, also known as proactive advising, involves advisors reaching out to students before problems arise. It is based on the premise that early intervention can prevent issues and improve student outcomes significantly (Earl, 1988).

Smith (2015) investigates developmental advising, which focuses on fostering both academic and personal growth. His study, using mixed methods, found that students who received developmental advising had a 20% higher retention and graduation rate than those who received prescriptive advising. Students reported feeling more supported and appreciated the holistic approach, emphasizing personalized advising. The study concluded that developmental advising enhances student satisfaction and academic outcomes by promoting advisor-student relationships. Johnson (2016) explores intrusive advising, or proactive advising, which involves reaching out to students before problems arise. His study found that students who received intrusive advising had a 15% higher retention rate. The research underscored the importance of early intervention and consistent advisor contact, which fosters a sense of accountability. The study recommended advisor training in intrusive advising techniques to improve student retention, especially for at-risk students.

Brown and Cooper (2017) focus on the integration of technology in academic advising. Their research found that tools like online scheduling, advising management systems, and communication platforms significantly improved advising efficiency and accessibility. While technology enhances the advising experience, the study highlighted challenges like technological barriers and the potential reduction of personal interaction. They suggested a hybrid advising model combining online and in-person support. Miller (2018) examines the advising challenges faced by first-generation college students. His study found that targeted interventions, such as mentoring programs and workshops, effectively supported these students, improving retention and graduation rates. The research emphasized the importance of building strong advisor-student relationships and providing culturally competent support tailored to the unique challenges of first-generation students. Overall, these studies underscore the importance of personalized and proactive advising in improving student retention, graduation, and satisfaction, while also highlighting the role of technology in enhancing advising practices.

Davis and Lee (2019) examined the impact of group advising sessions on student engagement and satisfaction at a mid-sized university. Group advising, which involves multiple students meeting with an advisor to discuss shared topics like course selection and career planning, proved to be an effective way to provide information to a large student population. The study employed both quantitative (student satisfaction surveys and engagement metrics) and qualitative methods (focus groups and interviews). Quantitative results revealed that group advising was generally well-received, with higher attendance and participation compared to individual sessions. Students valued the opportunity to learn from their peers' experiences, which fostered a sense of community. Qualitative data showed that group advising sessions helped students realize they were not alone in their concerns.

Despite its benefits, the study found that group advising lacked the personalized attention needed for specific issues, suggesting that it should be supplemented with individual sessions. Advisors emphasized the importance of strong facilitation skills to manage group dynamics and ensure that all students' concerns were addressed. The study concluded that group advising is an effective method for addressing common academic topics but recommended integrating individual advising for a more personalized approach to student support.

The literature on academic advising emphasizes its crucial role in student success, with various models positively influencing retention and academic outcomes. Developmental advising (Smith, 2015) enhances student persistence through strong advisor-student relationships and personalized support. Intrusive advising (Johnson, 2016) focuses on proactive intervention, improving retention by addressing issues early, especially for first-year students. The integration of technology in advising (Brown & Cooper, 2017) streamlines services, increasing accessibility and efficiency, but works best with in-person support.

First-generation students benefit from targeted advising strategies (Miller, 2018), including mentoring and academic workshops. Group advising (Davis & Lee, 2019) is effective for providing general information but requires individual

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 9, Issue 3, pp 13-27, March-2025

https://doi.org/10.47001/IRJIET/2025.903003

sessions for personalized support. Faculty advisors (Clark, 2020) play a critical role, but need adequate training and time. Online advising (Thompson, 2021) offers flexibility, though a hybrid model is more effective. Cultural competence (White & Harris, 2022) ensures advisors support diverse student populations. Peer advising (Lewis, 2023) can complement traditional advising, creating relatable guidance.

Despite these insights, gaps remain, including the long-term effects of advising models, the effectiveness of hybrid advising, and strategies to enhance cultural competence and sustain peer advising programs. This project aims to address these gaps and improve academic advising practices to better support student success in higher education.

III. METHODOLOGY ADOPTED

The development of the Student Advising System adopts the Object-Oriented Analysis and Design Method (OOADM). OOADM is a widely used methodology in software development that focuses on applying object-oriented principles to both the analysis and design of systems. This method is ideal for building complex systems that need to model real-world entities and interactions, making it suitable for the creation of a student advising platform. The methodology ensures the system is modular, scalable, maintainable, and easily extensible.

OOADM divides the software development process into two main stages: Object-Oriented Analysis (OOA) and Object-Oriented Design (OOD). The process begins by identifying the system's functional requirements and modeling real-world entities as objects. These objects are then designed and implemented with appropriate behaviors, attributes, and relationships in mind. This approach ensures that the system mimics real-world interactions, leading to intuitive and effective software solutions.

3.1 Analysis of Existing System

Existing student advising systems predominantly operate on manual processes, often requiring face-to-face interactions between advisors and students. These traditional models introduce several operational inefficiencies that can negatively affect both the quality and accessibility of advising services.

- **Inconvenience:** The reliance on in-person scheduling creates a significant barrier to timely communication, as students must physically visit advising offices to arrange appointments. This delay in securing consultations can lead to prolonged response times, particularly in high-demand periods, thereby impeding the overall advising process. Research indicates that this manual scheduling process not only exacerbates delays but also leads to frustration among students, thereby diminishing their satisfaction with academic advising services (Smith & Johnson, 2020).
- **Limited Accessibility:** Traditional advising systems often fail to accommodate students who are located in remote geographic regions or those who experience physical disabilities. For these students, the need to meet advisors in person can create substantial access barriers, limiting their ability to benefit from essential academic support. This issue is particularly critical as research highlights that students facing geographic or mobility challenges are at a higher risk for attrition and academic disengagement (Chen & Gray, 2018). Therefore, the limited accessibility of traditional advising systems may disproportionately impact at-risk student populations.
- **Inefficiency:** Manual advising systems also suffer from a lack of efficiency due to the labor-intensive nature of tracking student progress, scheduling, and record-keeping. Studies have shown that these processes often lead to administrative delays, errors in student records, and a misallocation of advisor time (Garrison & Kinzie, 2019). As a result, advisors may spend significant time on administrative tasks, which reduces the amount of time available for personalized, student-focused advising. This inefficiency further exacerbates the overall advising experience and limits the ability to provide tailored support to students.

In summary, the current reliance on manual, face-to-face advising systems presents multiple challenges that undermine the overall efficacy and accessibility of student support services. The existing literature suggests a growing need for technological interventions to address these barriers, enhance service efficiency, and improve access to academic advising for all students, particularly those from underserved or disadvantaged backgrounds.

3.2 Data Flow of the Existing System

In the existing system, data flow is often fragmented and manual, with students manually providing course selections, advisors recording progress by hand, and administrative tasks handled through disparate systems.

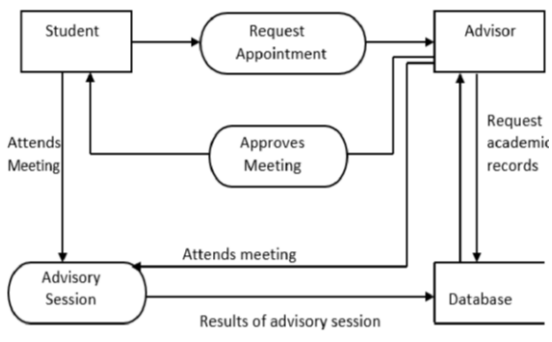


Figure 3.1: Data Flow of the Existing System

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 9, Issue 3, pp 13-27, March-2025

https://doi.org/10.47001/IRJIET/2025.903003

3.3 Analysis of the Proposed System

The proposed Student Advising System aims to solve the problems identified in the existing system by providing a web-based platform where students can interact with their advisors in realtime, view available courses, enroll, and track their academic progress. The system is designed to be user-friendly, efficient, and scalable, capable of serving a large number of students and advisors. With the following advantages: The implementation of an automated advising system offers significant benefits to both students and advisors. By enabling students to access the system from anywhere at any time, it enhances convenience and flexibility in communication, allowing for smoother course enrollment and ongoing academic support. This remote access ensures that advising services are available to all students, regardless of geographic location or physical limitations, promoting greater inclusivity.

Furthermore, automation of key administrative tasks—such as course enrollment, advisor assignment, and tracking of academic progress—reduces the risk of human error and accelerates the advising process. This leads to more efficient operations, ensuring that students receive timely feedback from advisors, which helps them make well-informed decisions about their academic paths. Additionally, by automating routine functions, advisors can focus more on addressing complex student needs, optimizing their time and resources. Overall, the adoption of such systems not only increases the accessibility and efficiency of advising but also contributes to a more streamlined, equitable, and effective academic advising experience for all parties involved.

3.4 Data Flow Diagram of the Proposed System

The Data Flow Diagram (DFD) shows how data is processed and flows between different parts of the system, from student logins to course enrollments and advisor communications.

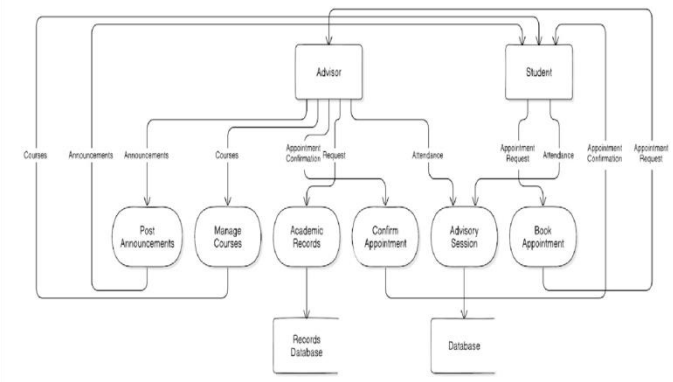


Figure 3.2: Data Flow Diagram of the Proposed System

3.5 Unified Modeling Language (UML)

Unified Modeling Language (UML) diagrams are used to model the structure and behavior of the proposed system.

3.5.1 Use Case Diagram: Captures interactions between students, advisors, and the system, including course enrollment and academic advising.

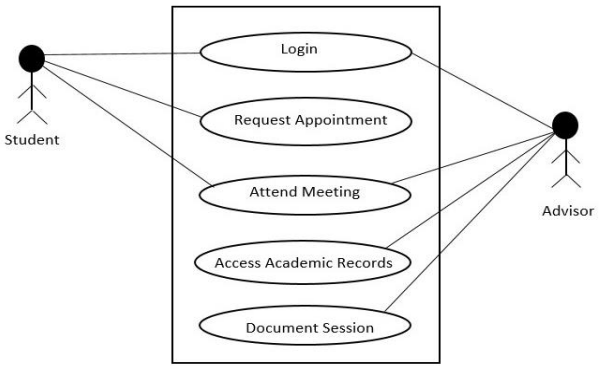


Figure 3.3: Use Case Diagram of the Proposed System

- Class Diagram: Shows the system's structure, including key objects like `Student`, `Advisor`, and `Course`, along with their attributes and methods.

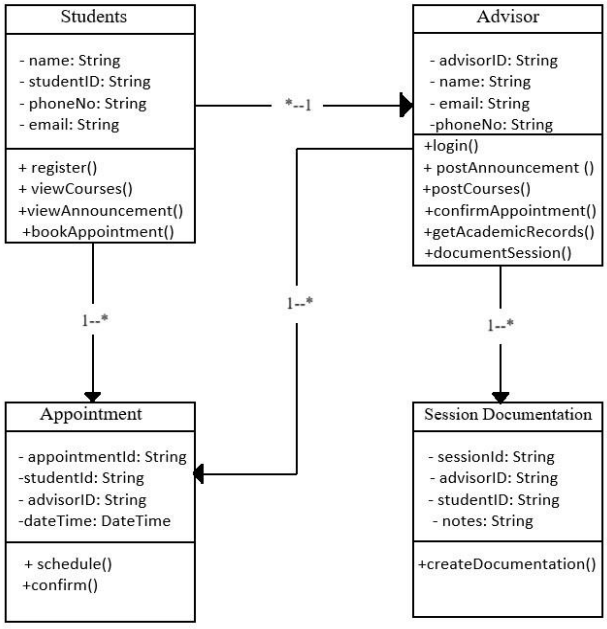


Figure 3.4: Class Diagram of the Proposed System

- Activity Diagram: An activity diagram is a behavioral diagram in Unified Modeling Language (UML) that visualizes the workflow of a system, process, or use case. It illustrates the sequence of activities, actions, and decisions involved in a process, from start to finish.

International Research Journal of Innovations in Engineering and Technology (IRJIET)
ISSN (online): 2581-3048
Volume 9, Issue 3, pp 13-27, March-2025
https://doi.org/10.47001/IRJIET/2025.903003

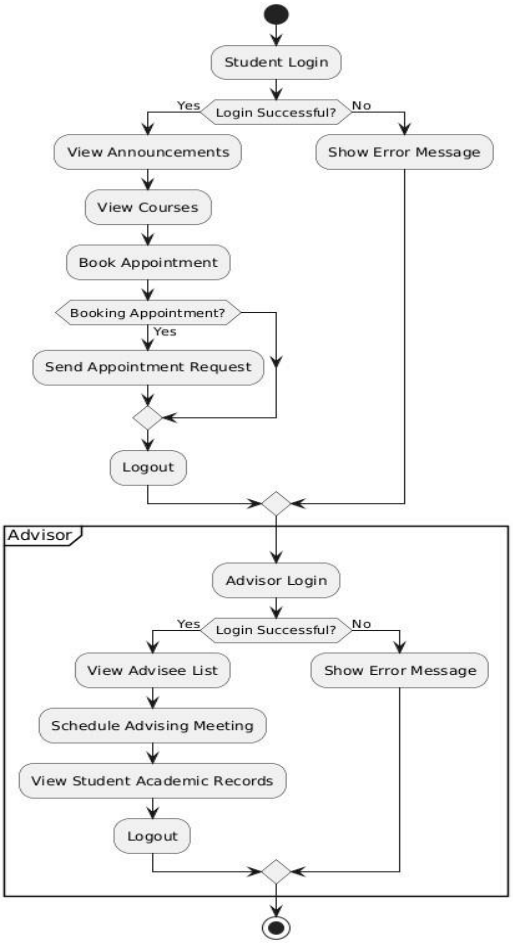


Figure 3.5: Activity Diagram of the Proposed System

3.6 High-Level Model of the Proposed System

The high-level model outlines the structure of the proposed system, focusing on the interactions between the user interface, the business logic, and the database.

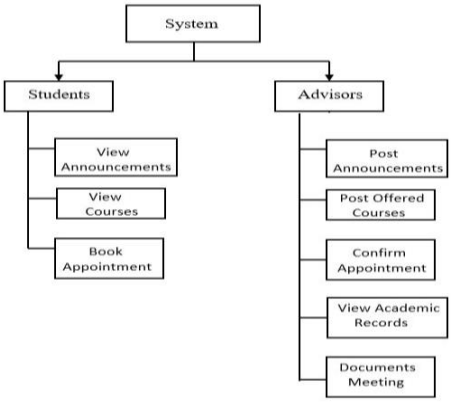


Figure 3.6: High-Level Model of the Proposed System

IV. OBJECTIVES OF THE DESIGN

The Student Advising System is designed to streamline communication and interactions between students and their advisors, this system aims to provide students with an efficient way to access announcements, book appointments, and allow advisors to view students' academic grades. The primary objectives of the system are:

- **Centralized platform for announcements:** Provide students with a single place to view important announcements from advisors about academic and administrative matters.
- **Appointment booking:** Allow students to easily schedule meetings with their advisors for academic discussions or guidance.
- **Access to academic records:** Enable advisors to view students' grades, helping them offer more personalized advice.
- **User-friendly design:** Ensure the system is easy to navigate and accessible for both students and advisors.
- **Security:** Securely manage students' academic records for personalized advising.

4.1 Control Centre/Main Menu

The Main Menu serves as the central hub of the Students Advising System, granting users access to the system's core functionalities. The layout prioritizes simplicity and clarity to ensure that users can easily navigate the system without prior technical knowledge. Figure 4.1 below illustrates the main menu's layout.

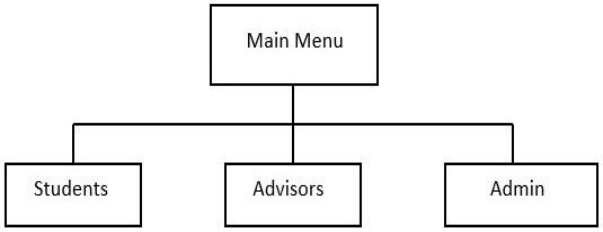


Figure 4.1: Main Menu Flow Diagram

4.2 Students Subsystem

The Students subsystem handles the process of adding your level advisor, viewing announcements and also booking appointments to see the advisor.

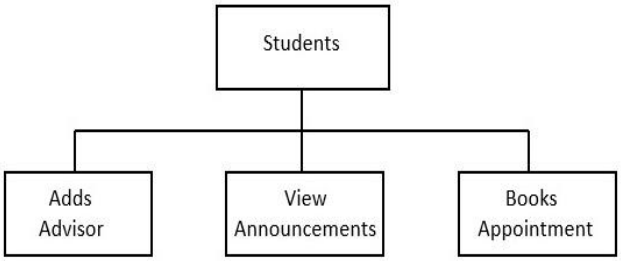


Figure 4.2: Students Flow Diagram

International Research Journal of Innovations in Engineering and Technology (IRJIET)
ISSN (online): 2581-3048
Volume 9, Issue 3, pp 13-27, March-2025
https://doi.org/10.47001/IRJIET/2025.903003

4.3 Advisors Subsystem

The Advisors Subsystem simplifies the process of confirming appointments and sending announcements. Students can send appointment requests and the advisor confirms the date that fits their schedule, the student is notified after the confirmation.

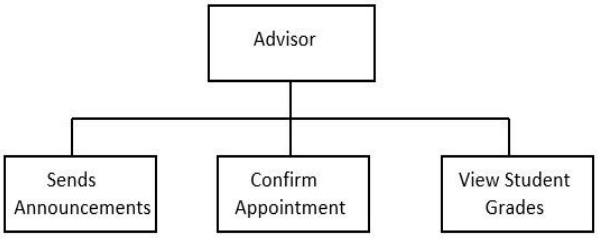


Figure 4.3: Appointment Booking Flow Diagram

4.4 Admin Subsystem

The Admin Subsystem is designed for system administrators to manage user accounts, updates students' grades, oversee system performance, and ensure the smooth operation of all functionalities. Administrators can add or remove users, update system settings, and monitor the overall health of the system.

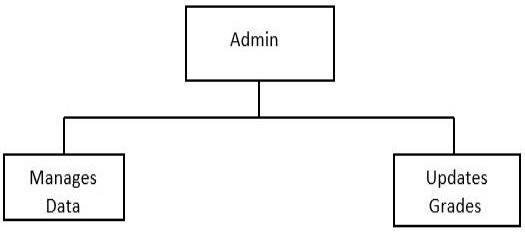


Figure 4.4: Admin Flow Diagram

4.5 Database Development Tool

The database development tool used in this project is the Firestore Database. Firestore Database, developed by Google, is a NoSQL, document-oriented database designed for scalable and high performance applications. It provides a flexible and efficient way to manage data, which is critical for applications requiring real-time updates and scalable solutions. Firestore Database utilizes a JSON-like data model for storing data, which differs significantly from traditional relational databases. In Firestore, data is organized into collections and documents, rather than tables with fixed schemas. Each document contains fields with key-value pairs, and these documents are grouped into collections. This model allows for a highly flexible and hierarchical data structure that can adapt to various types of data.

4.6 Database Design and Structure

The database is structured to support the system's various functionalities, with distinct tables for users, announcements, appointments, and academic records. Each table is designed with specific fields that capture all necessary data points while maintaining normalization to prevent data redundancy.

Table 4.1: Users Table Structure

Field Name	Data Type	Description
user_id	Int	Unique identifier for each user
Name	String	Full name of the user
Email	String	User's email address
password	String	Hashed password for authentication
Role	String	Role of the user (student, advisor, admin)
created_at	Timestamp	Account creation timestamp
updated_at	Timestamp	Last update timestamp

Table 4.2: Announcements Table Structure

Field Name	Data Type	Description
announcement_id	Int	Unique identifier for each announcement
advisor_id	Int	Identifier for the advisor who made the announcement
title	String	Title of the announcement
content	String	Content of the announcement
created_at	Timestamp	Timestamp when the announcement was created
updated_at	Timestamp	Timestamp when the announcement was last updated

International Research Journal of Innovations in Engineering and Technology (IRJIET)
ISSN (online): 2581-3048
Volume 9, Issue 3, pp 13-27, March-2025
https://doi.org/10.47001/IRJIET/2025.903003

Table 4.3: Appointments Table Structure

Field Name	Data Type	Description
appointment_id	Int	Unique identifier for each appointment
student_id	Int	Identifier for the student booking the appointment
advisor_id	Int	Identifier for the advisor with whom the appointment is booked
appointment_date	DateTime	Date and time of the appointment
status	String	Status of the appointment (e.g., confirmed, cancelled)
created_at	Timestamp	Timestamp when the appointment was created
updated_at	Timestamp	Timestamp when the appointment was last updated

Table 4.4: Grades Table Structure

Field Name	Data Type	Description
grade_id	Int	Unique identifier for each grade record
student_id	Int	Identifier for the student
course_code	String	Course code for the related subject
grade	String	Grade achieved by the student
semester	String	Semester for which the grade is recorded
created_at	Timestamp	Timestamp when the grade record was created
updated_at	Timestamp	Timestamp when the grade record was last updated

4.7 Input/Output Format

The input/output format is designed to facilitate user interactions with the system. Inputs include data from forms such as login credentials, appointment requests, and announcements. Outputs include confirmation messages, announcements displayed on the dashboard, and scheduled appointments.

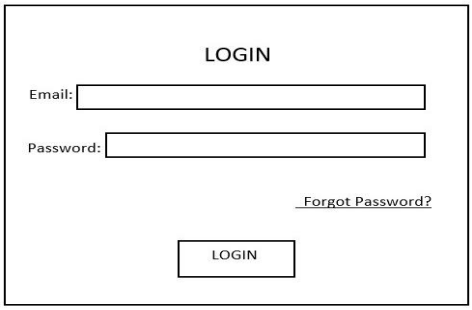


Figure 4.5: Log-in Page

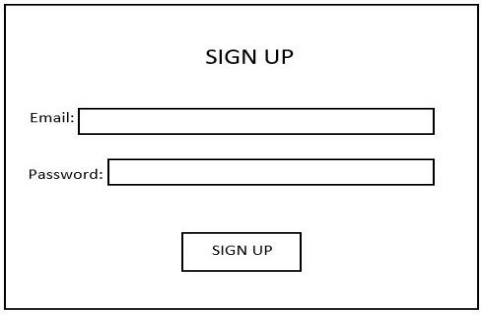


Figure 4.6: Sign-up Page

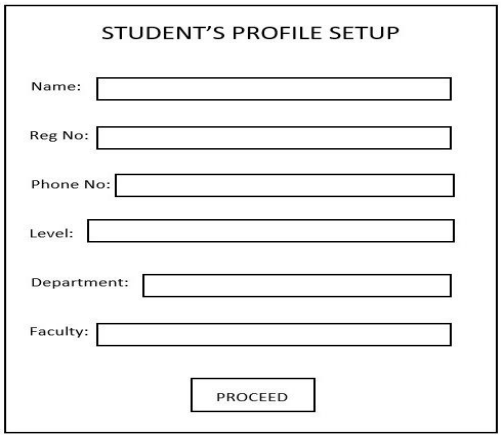


Figure 4.7: Student Profile Setup Page

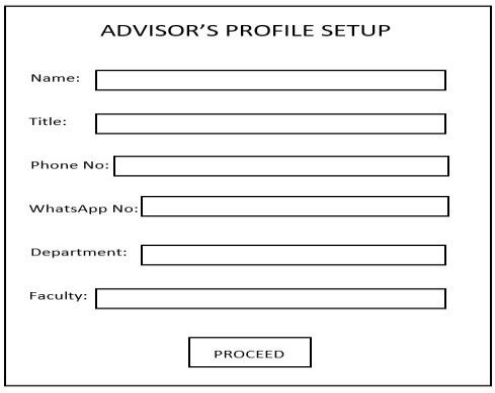


Figure 4.8: Advisor Profile Setup Page

International Research Journal of Innovations in Engineering and Technology (IRJIET)
ISSN (online): 2581-3048
Volume 9, Issue 3, pp 13-27, March-2025
https://doi.org/10.47001/IRJIET/2025.903003

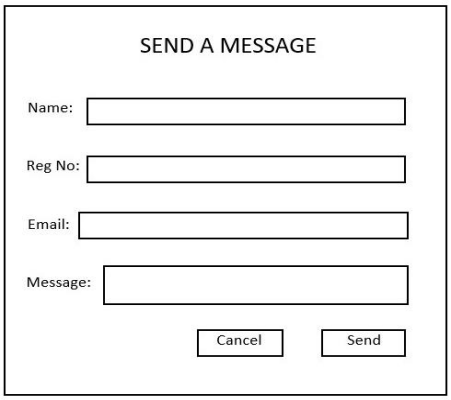


Figure 4.9: Send a Message to Advisor Page

4.8 Algorithm

The algorithm describes the step-by-step procedures for key operations within the system, ensuring that processes are executed accurately and efficiently. Below are algorithms for creating new announcements and booking appointments.

4.8.1 Algorithm for Posting Announcement

1. Function post Announcement (advisorId, title, content)
2. Begin
3. If advisorId is empty OR title is empty OR content is empty
4. Then
5. Throw Error 'All fields are required.'
6. End If
7. Save announcement to the database with the following details:
   - advisor_id: advisorId
   - title: title
   - content: content
   - created_at: current date and time
8. If announcement is saved successfully
9. Then
10. Print 'Announcement posted successfully.'
11. Return { success: true, announcement }
12. Else
13. Print 'Error posting announcement:' + error message
14. Return { success: false, error: error message }
15. End

4.8.2 Algorithm for Booking Appointment Pseudo code

1. Function book Appointment(studentId, advisorId, appointmentDate)
2. Begin
3. If studentId is empty OR advisorId is empty OR appointmentDate is empty
4. Then
5. Throw Error 'All fields are required.'
6. End If
7. Save appointment details to the database with the following details:
   - student_id: studentId
   - advisor_id: advisorId
   - appointment_date: appointmentDate
   - status: 'confirmed'
   - created_at: current date and time
8. If appointment is saved successfully
9. Then
10. Print 'Appointment booked successfully.'
11. Return { success: true, appointment }
12. Else
13. Print 'Error booking appointment:' + error message
14. Return { success: false, error: error message }
15. End

4.9 Program Flowchart

A program flowchart provides a visual representation of the workflow and processes within the system. It depicts the sequence of operations, decisions, and data flow.

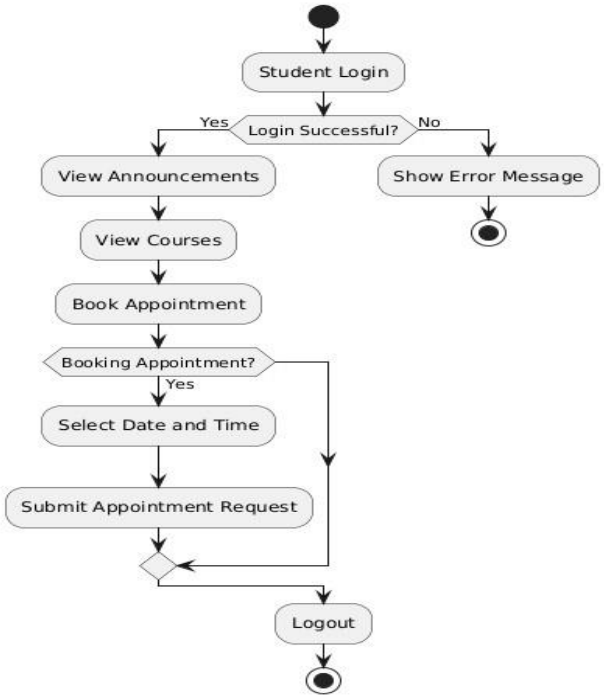


Figure 4.10: Program Flowchart

4.10 Hardware Requirements

The hardware required for effectively running the system includes:

- **An Android Smartphone:** For testing mobile responsiveness and functionality.

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 9, Issue 3, pp 13-27, March-2025

https://doi.org/10.47001/IRJIET/2025.903003

- **Dual/Quad or Octa-Core Processor of 1.60 GHz and Above:** Ensures the system's performance is optimal and can handle multiple tasks simultaneously.
- **2 GB RAM Memory and Above:** Minimum RAM requirement to ensure smooth operation of the application.
- **Storage Capacity of 8 GB and Above:** To accommodate the operating system, development tools, and application data.
- **At Least 1 GB of Free Storage Capacity:** For temporary files and application updates.

4.11 Software Requirements

The software requirements for the system include:

- **Operating System:** Windows 7 or higher is recommended to ensure compatibility with development tools and libraries.
- **Visual Studio Code (VS Code):** The primary IDE for writing and debugging the application's code.
- **Node.js:** A runtime environment for executing JavaScript code on the server-side. It is essential for backend development and package management using npm.
- **React.js:** A JavaScript library for building user interfaces, used for the frontend development.
- **Firestore Database:** A NoSQL database hosted on Firebase Console for storing and managing application data.
- **GitHub:** For version control and code management, facilitating collaboration and tracking changes.

4.12 Choice of Programming Environment

The choice of programming environment includes:

- **Visual Studio Code:** Selected for its powerful code editing features, extensions for React.js and Node.js, and integrated terminal for running scripts and commands.
- **Node.js and React.js:** Node.js provides a runtime environment for server-side development and managing dependencies with npm. React.js offers a flexible and efficient library for building dynamic user interfaces. The combination allows for a fullstack JavaScript development approach, simplifying the development process and improving productivity.

V. SUMMARY AND CONCLUSION

This project, titled "Design and Development of a Web-Based Student Academic Advising System," addresses the inefficiencies inherent in traditional academic advising processes in higher education. The objective was to design and implement a system that automates and optimizes various manual tasks within the advising workflow, thereby enhancing communication and interaction between students and academic advisors. The system offers functionalities such as student access to academic records, tracking of academic progress, automated course registration recommendations, and scheduling of advising appointments. Academic advisors, on the other hand, can manage student profiles, monitor performance metrics, provide academic guidance, and generate performance reports.

The system was developed utilizing modern technologies, including React.js and Tailwind CSS for the frontend interface, and Node.js with Firebase for backend operations. The platform is designed to be accessible via a web browser, ensuring seamless access for both students and advisors, regardless of geographic location, provided internet connectivity is available. Rigorous testing and evaluation revealed that the system significantly reduced the administrative burden on advisors by automating routine tasks such as prerequisite checks and performance monitoring. Furthermore, students reported increased satisfaction with the advising process, citing the system's user-friendly interface, faster response times, and improved access to relevant academic information.

The system effectively addresses several challenges posed by traditional advising practices, such as the excessive time spent on administrative tasks, limited accessibility to timely academic information, and the difficulty of providing individualized advising. By automating these routine processes, the system enables academic advisors to allocate more time for personalized academic counseling. Additionally, the system empowers students by providing them with greater control over their academic planning, allowing for more informed decisions regarding course selection and long-term academic goals. The web-based nature of the system ensures continuous accessibility, which is essential in the modern educational landscape, where students and faculty may not always be physically present on campus.

In conclusion, the implementation of this web-based advising system significantly enhances the academic advising experience for both students and advisors. It contributes to the efficiency, accuracy, and timeliness of the advising process, which ultimately fosters improved academic performance and overall student success.

REFERENCES

[1] Anusiuba, O. I. A (2024), An Automated and Robust Tertiary Institution Web-Based Student Complaint Management System, International Research Journal of Innovations in Engineering and Technology (IRJIET),

ISSN (online): 2581-3048
Volume 9, Issue 3, pp 13-27, March-2025
https://doi.org/10.47001/IRJIET/2025.903003

7(12), 110-124. https://doi.org/10.47001/IRJIET/2023.712016.

[2] Anusiuba O.I.A., Nweke N. M., Egbo, F. N.,(2019). Improving Students' Achievement And Retention In Computer Studies Through The Use Of Computer-Assisted Tutorial Instructions, International Journal of Innovative Research and Advanced Studies (IJIRAS) 6 (Issue 8) 85-91.

[3] Anusiuba O. I. A, Karim Usman, Ekwealor O. U, Igbonekwu J. O.,(2021). Design and Implementation of a Tertiary Institution Web-Based Student Complaint Management System, International Journal of Software & Hardware Research in Engineering (IJSHRE), 9(8), 23-36.

[4] Bates, A. W., &Sangra, A. (2011). Managing technology in higher education: Strategies for transforming teaching and learning. Jossey-Bass. Retrieved from https://brooks.google.com

[5] Brown, S., & Cooper, D. (2017). The role of technology in academic advising. Journal of Academic Advising, 20(3), 145-162.Retrieved from https://www.jstor.org

[6] Chickering, A. W.,& Reisser, L. (1993). Education and identity. Jossey-Bass. Retrieved from https://books.google.com

[7] Clark, M. (2020). The role of faculty advisors in academic advising. Higher Education Quarterly, 74(2), 289-305.

[8] Crookston, B. B. (1972). A developmental view of academic advising as teaching. Journal of College Student Personnel, 13(1), 12-17. Retrieved from https://www.nacada.ksu.edu

[9] Davis, J., & Lee, A. (2019). The effectiveness of group advising sessions. Advising Today, 24(1), 37-49.

[10] Earl, W. R. (1988). Intrusive advising of freshmen in academic difficulty. NACADA Journal, 8(2), 27-33.

[11] Johnson, L. (2016). Intrusive advising and student retention. Journal of Student Success and Retention, 12(2), 23-45.

[12] Kalsbeek, D. (2006). The role of academic advising in the student retention process. The Journal of College Student Retention: Research, Theory & Practice, 8(1), 61-73.

[13] Kuh, G. D. (2008). High-impact educational practices: What they are, who has access to them, and why they matter. Association of American Colleges and Universities. Retrieved from https://www.aacu.org

[14] Lewis, T. (2023). The impact of peer advising programs. Journal of Higher Education, 18(1), 77-94.

[15] Miller, R. (2018). Advising strategies for first-generation college students. Journal of FirstGeneration Student Success, 4(2), 56-74.

[16] Sanford, N. (1966). Self & society: Social change and individual development. Atherton Press. Retrieved from https://archive.org

[17] Selwyn, N. (2013). Education in a digital world: Global perspectives on technology and education. Routledge.

[18] Smith, J. (2015). The impact of developmental advising on student success. Academic Advising Journal, 21(1), 67-89. Retrieved from https://www.nacadajournal.org

[19] Thompson, P. (2021). Online academic advising in the era of COVID-19. Journal of Online Learning and Teaching, 17(2), 134-150.

[20] Tinto, V. (1993). Leaving college: Rethinking the causes and cures of student attrition. University of Chicago Press. Retrieved from https://press.uchicago.edu

[21] Tinto, V. (2012). Completing college: Rethinking institutional action. University of Chicago Press. Retrieved from https://press.uchicago.edu

[22] White, A., & Harris, R. (2022). Advising minority students: Challenges and best practices. Journal of Multicultural Counseling and Development, 40(3), 210-225.

Citation of this Article:

Umeh, Mary Rose Ngozi, Anusiuba, Overcomer Ifeanyi Alex, Faseki, Ngozi Francisca, & Davies, Naomi Nmesoma. (2025). Design and Implementation of an Advanced Web-Based Academic Advising System for Optimizing Student Support and Engagement. *International Research Journal of Innovations in Engineering and Technology - IRJIET*, 9(3), 13-27. Article DOI https://doi.org/10.47001/IRJIET/2025.903003

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 9, Issue 3, pp 13-27, March-2025

https://doi.org/10.47001/IRJIET/2025.903003

APPENDIX A

Program Listing

```
// Function to post a new announcement async function post Announcement (advisorId, title, content) {    try {
      // Validate inputs       if (!advisorId || !title || !content) {           throw new Error('All fields are required.');
      }
      // Save announcement to the database    const announcement = await db.collection('announcements').add({           advisor_id:
advisorId,     title: title,  content: content,    created_at: new Date(),
      });
console.log('Announcement posted successfully.');       return { success: true, announcement };
   } catch (error) {        console.error('Error posting announcement:', error.message);    return { success: false, error: error.message
};
   }
}
```

```
// Function to book an appointment
async function bookAppointment(studentId, advisorId, appointmentDate) {    try {
      // Validate inputs
if (!studentId || !advisorId || !appointmentDate) {
          throw new Error('All fields are required.');
      }
      // Save appointment details to the database const appointment = await db.collection('appointments').add({student_id:
studentId, advisor_id: advisorId, appointment_date: appointmentDate, status: 'confirmed', created_at: new Date(),
      });
console.log('Appointment booked successfully.');       return { success: true, appointment };
   } catch (error) {       console.error('Error booking appointment:', error.message);       return { success: false, error: error.message
};
   }
}
```

```
// Function to create a new user async function createUser(email, password, userDetails) {    try {
      // Validate user input       if (!email || !password) { throw new Error('Email and password are required.');
      }
      // Create a new user with Firebase Authentication                constuserCredential = await
firebase.auth().createUserWithEmailAndPassword(email, password);       const user = userCredential.user;

// Save additional user details to Firestore       await firebase.firestore().collection('users').doc(user.uid).set({           email: email,
...userDetails
      });
      // Display success message       console.log('User successfully registered.');       return { success: true, user };
   } catch (error) {       // Handle errors       console.error('Error creating user:', error.message);       return { success: false, error:
error.message };
   }
}
```

```
// Function to log in a user async function loginUser(email, password) {
   try {
      // Validate user input       if (!email || !password) {           throw new Error('Email and password are required.');
      }
      // Sign in the user with Firebase Authentication                constuserCredential = await
firebase.auth().signInWithEmailAndPassword(email, password);       const user = userCredential.user;
```


International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

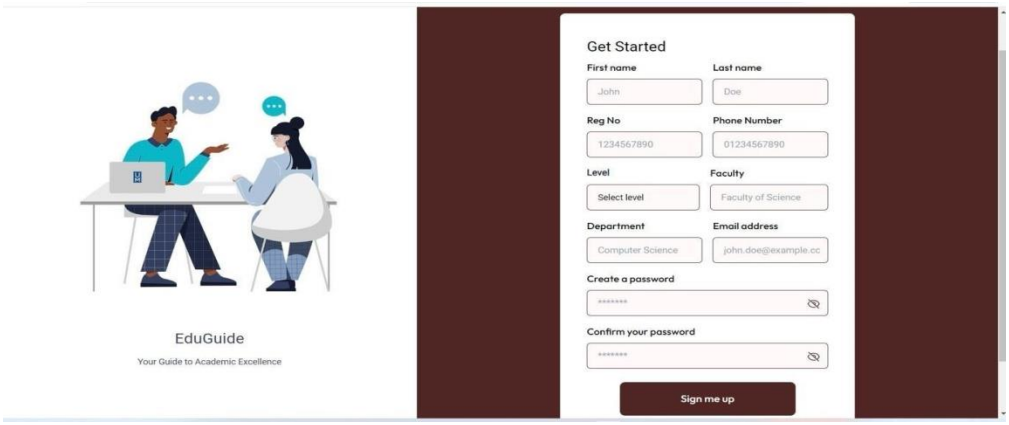
Volume 9, Issue 3, pp 13-27, March-2025

https://doi.org/10.47001/IRJIET/2025.903003

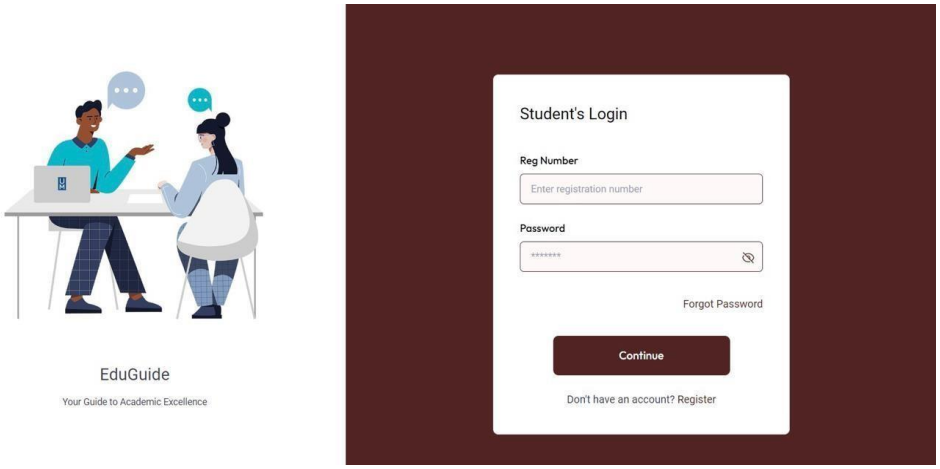
```
      // Display success message
console.log('User successfully logged in.');       return { success: true, user };
   } catch (error) {       // Handle errors       console.error('Error logging in user:', error.message);       return { success: false, error:
error.message };
   }
}
```

APPENDIX B

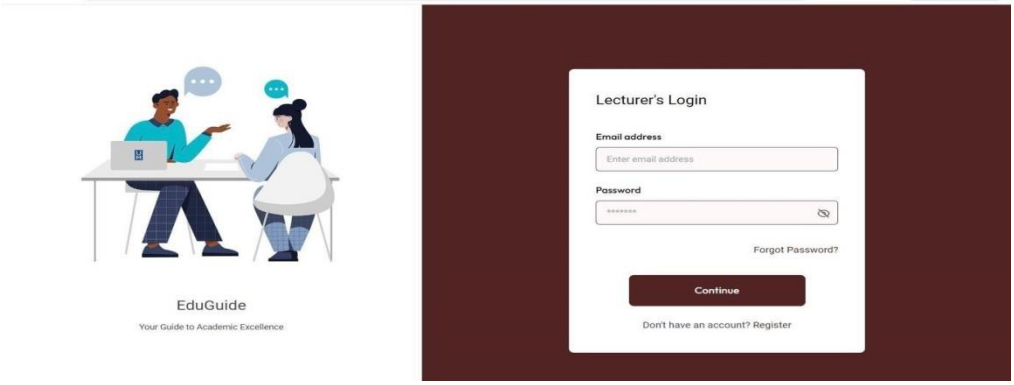
Sample Output



Student Profile Setup Page



Student Log-in Page



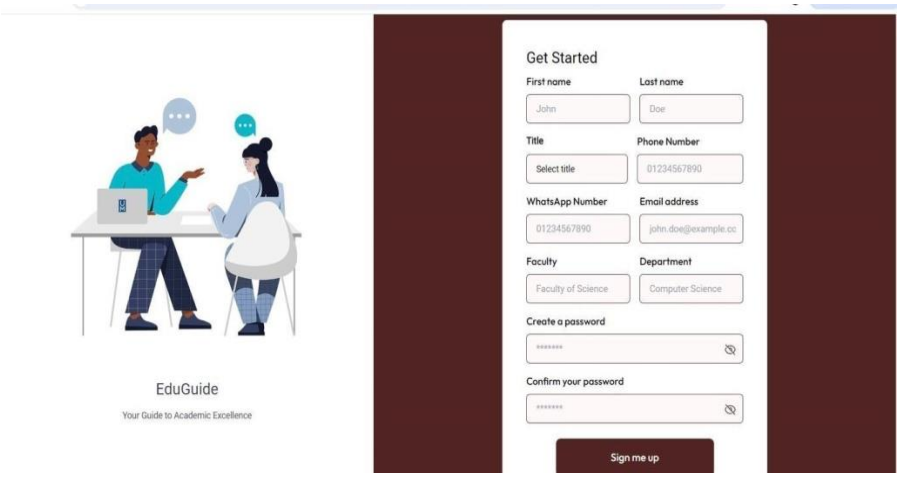
International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

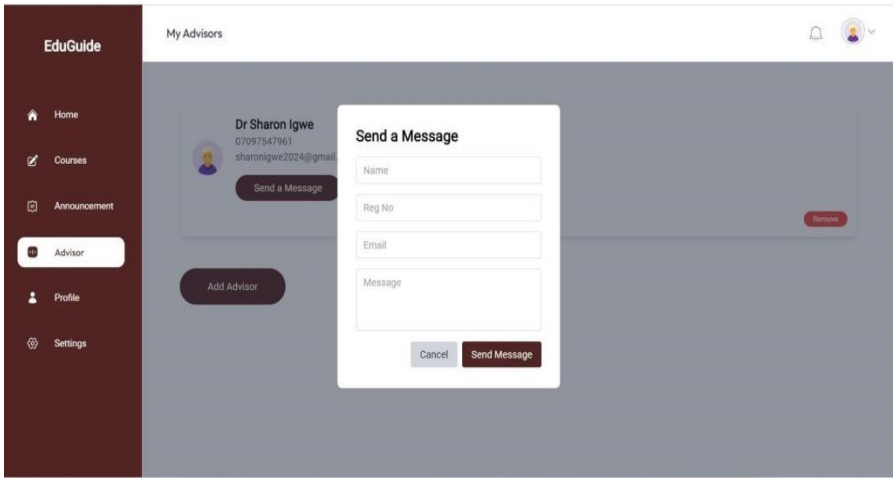
Volume 9, Issue 3, pp 13-27, March-2025

https://doi.org/10.47001/IRJIET/2025.903003

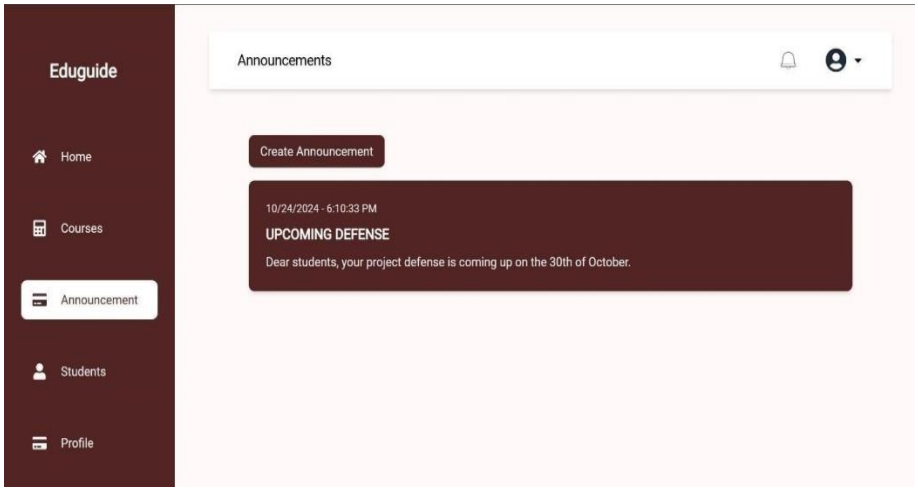
Advisor Log-in Page



Advisor Profile Setup Page

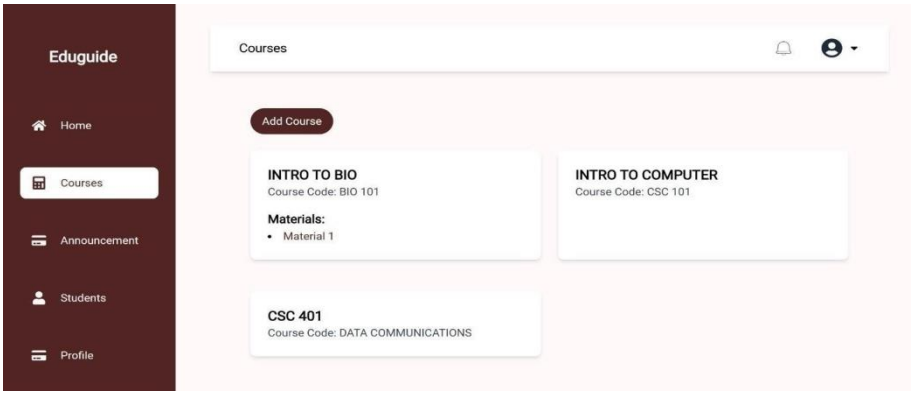


Send A Message to Advisor Page



International Research Journal of Innovations in Engineering and Technology (IRJIET)
ISSN (online): 2581-3048
Volume 9, Issue 3, pp 13-27, March-2025
https://doi.org/10.47001/IRJIET/2025.903003

Create Announcement Page



****

